Numerical integration over 2D NURBS-shaped domains with applications to NURBS-enhanced FEM

Ruben Sevilla, Sonia Fernández-Méndez *

Laboratori de Càlcul Numèric (www-lacan.upc.edu), Departament de Matemàtica Aplicada III, E.T.S. de Ingenieros de Caminos, Canales y Puertos, Universitat Politècnica de Catalunya, Jordi Girona 1, E-08034 Barcelona, Spain

A B S T R A C T

This paper focuses on the numerical integration of polynomial functions along non-uniform rational B-splines (NURBS) curves and over 2D NURBS-shaped domains, i.e. domains with NURBS boundaries. The integration of the constant function $f=1$ is of special interest in computer aided design software and the integration of very high-order polynomials is a key aspect in the recently proposed NURBS-enhanced finite element method (NEFEM). Several well-known numerical quadratures are compared for the integration of polynomials along NURBS curves, and two transformations for the definition of numerical quadratures in triangles with one edge defined by a trimmed NURBS are proposed, analyzed and compared. When exact integration is feasible, explicit formulas for the selection of the number of integration points are deduced. Numerical examples show the influence of the number of integration points in NEFEM computations.

1. Introduction

Non-uniform rational B-splines (NURBS) [1] are widely used in computer aided design (CAD). Some basic tools of CAD software are the computation of the length of a NURBS curve, the subdivision of a NURBS curve in equally spaced pieces and the computation of the area of a domain with NURBS boundaries, to name a few. These basic operations require the numerical integration of the constant function $f=1$ along NURBS curves and over domains with NURBS boundaries.

On the other hand, CAD models are usually employed by the finite element (FE) community in the preprocess stage, in order to build a spatial discretization of the computational domain. Once the discretization is generated, the exact boundary representation is replaced by a piecewise polynomial approximation. However, in the last decade many authors have pointed out the importance of the geometrical model in FE simulations, see for instance [2–5]. This fact has motivated novel numerical methodologies considering exact CAD descriptions of the computational domain. For instance, NURBS-enhanced finite element method (NEFEM) considers an exact representation of the geometry while maintaining the standard polynomial approximation of the solution. With the NEFEM approach standard FE interpolation and numerical integration is used in the large majority of the domain (i.e., in the interior, for elements not intersecting the boundary) preserving the computational efficiency of classical FE techniques. Specifically designed piecewise polynomial interpolation and numerical integration is required for those FEs along the NURBS boundary.

This paper is devoted to the study of the numerical integration of low- and high-order polynomial functions along trimmed NURBS curves and the integration over curved triangular elements with one edge defined by a trimmed NURBS. Particular emphasis is placed in the numerical integration of high-order polynomials, with applications to NEFEM. Several numerical quadratures are proposed and compared through numerical examples. The generalization to 3D domains is conceptually easy but it requires some extra attention to geometrical aspects and it is presented in [6].

Sections 2 and 3 recall the basic concepts on NURBS and NEFEM in two dimensions. Section 4 is devoted to the integration along NURBS curves. Some well-known 1D numerical quadratures are tested for the numerical integration of low- and high-order polynomials. The integration over triangular elements with one edge defined by a trimmed NURBS is addressed in Section 5. Two transformations for the definition of a numerical quadrature over a curved triangle are considered. The first one is a transformation from a straight-sided triangle in order to test the performance of triangle quadratures. The second one is a transformation from a rectangle to the curved triangle. When exact integration is feasible, explicit formulas for the selection of the number of integration points are deduced. Finally, numerical examples in

* Corresponding author. Tel.: +34 934011625; fax: +34 934011825.
E-mail addresses: ruben.sevilla@upc.edu (R. Sevilla), sonia.fernandez@upc.edu (S. Fernández-Méndez).

Section 6 show the influence of the number of integration points in NEFEM computation.

2. Basic concepts on NURBS curves

This section presents some basic notions of NURBS curves in order to introduce the notation and the concepts employed in the following sections. For a detailed presentation see for instance [1].

A q-th degree NURBS curve is a piecewise rational function defined in parametric form as

$$\boldsymbol{C}(\lambda) = \left(\sum_{i=0}^{\mathrm{n_{cp}}} v_i \boldsymbol{B}_i C_{i,q}(\lambda) \right) \Big/ \left(\sum_{i=0}^{\mathrm{n_{cp}}} v_i C_{i,q}(\lambda) \right), \quad \lambda \in [\lambda_a, \lambda_b], \tag{1}$$

where $\{\boldsymbol{B}_i\}$ are the coordinates of the *control points* (determining the *control polygon*), $\{v_i\}$ are their control weights, $\{C_{i,q}(\lambda)\}$ are the B-spline basis functions of degree q, and the interval $[\lambda_a, \lambda_b]$ is called the *parametric space*. The B-spline basis functions are defined recursively from the so-called *knot vector* $\Lambda = \{\lambda_0, \ldots, \lambda_{\mathrm{n_k}}\} = \{\underbrace{\lambda_a, \ldots, \lambda_a}_{q+1}, \lambda_{q+1}, \ldots, \lambda_{\mathrm{n_k}-q-1}, \underbrace{\lambda_b, \ldots, \lambda_b}_{q+1}\}$ by

$$C_{i,0}(\lambda) = \begin{cases} 1, & \lambda \in [\lambda_i, \lambda_{i+1}), \\ 0, & \lambda \notin [\lambda_i, \lambda_{i+1}), \end{cases} \tag{2}$$

$$C_{i,k}(\lambda) = \frac{\lambda - \lambda_i}{\lambda_{i+k} - \lambda_i} C_{i,k-1}(\lambda) + \frac{\lambda_{i+k+1} - \lambda}{\lambda_{i+k+1} - \lambda_{i+1}} C_{i+1,k-1}(\lambda), \tag{3}$$

for $k = 1 \ldots q$, where λ_i, for $i = 0, \ldots, \mathrm{n_k}$, are the *knots* or *breakpoints*. Note that the first and final knots must coincide with the endpoints of the parametrization interval and their multiplicity is always $q+1$. The multiplicity of the remaining knots, when it is larger than one, determines the decrease in the number of continuous derivatives. The number of control points, $\mathrm{n_{cp}}+1$, and knots, $\mathrm{n_k}+1$, are related to the degree of the parametrization, q, by the relation $\mathrm{n_k} = \mathrm{n_{cp}} + q + 1$, see [1] for more details.

It is worth remarking that $\sum_{i=0}^{\mathrm{n_{cp}}} C_{i,q}(\lambda) = 1$. Thus, Eq. (1) reduces to a (polynomial) B-spline curve when all the control weights are equal.

To summarize, a NURBS is just a piecewise rational function whose definition changes at breakpoints. Fig. 1 shows a NURBS curve and its control polygon.

In practice, CAD manipulators work with *trimmed* NURBS, which are defined as the initial parametrization restricted to a subspace of the parametric space. Fig. 2 shows the NURBS curve represented in Fig. 1 trimmed to the subinterval [0.05,0.75].

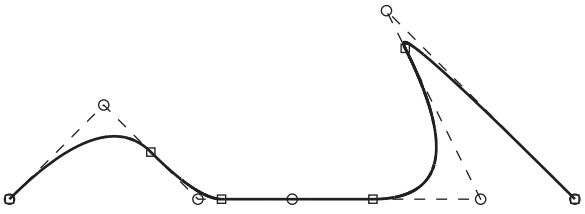

Fig. 1. NURBS curve (solid line), control points (○), control polygon (dashed line) and image of the breakpoints (□).

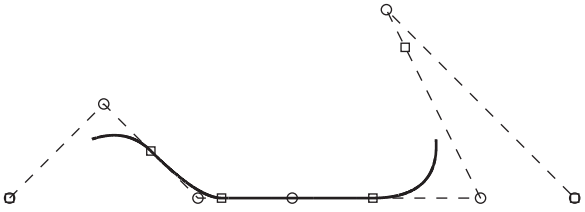

Fig. 2. Trimmed NURBS curve for $\lambda \in [0.05, 0.75]$.

3. NEFEM fundamentals

Let $\Omega \subset \mathbb{R}^2$ be an open bounded domain whose boundary $\partial\Omega$ or a portion of it, is curved. A regular partition of the domain $\overline{\Omega} = \bigcup_e \overline{\Omega}_e$ in subdomains, triangles in this work, is assumed, such that $\Omega_i \bigcap \Omega_j = \emptyset$, for $i \neq j$. It is important to remark that, in the following, Ω_e denotes the element with an exact description of the curved boundary. For instance, Fig. 3 shows a domain with part of the boundary described by a NURBS curve corresponding to an airfoil profile, and a triangulation of the domain with curved FEs with an exact boundary representation, i.e. curved *NEFEM elements*.

As usual in FE mesh generation codes, it is assumed that every curved boundary edge belongs to a unique NURBS. That is, one element edge cannot be defined by portions of two (or more) different NURBS curves. But on the contrary, it is important to note that breakpoints, which characterize the piecewise nature of NURBS, are independent of the mesh discretization. Thus, the NURBS parametrization can change its definition inside one edge, that is breakpoints may belong to element edges and do not need to coincide with FE nodes.

Every *interior* element (i.e. elements not having an edge that coincides with the NURBS boundary) can be defined and treated as standard FEs. Therefore, in the vast majority of the domain, interpolation and numerical integration are standard. For elements with at least one edge on the NURBS boundary a specifically designed interpolation and numerical integration is considered.

The polynomial approximation is defined with the Cartesian coordinates $\boldsymbol{x}$,

$$u(\boldsymbol{x}) \simeq u^h(\boldsymbol{x}) = \sum_{i=1}^{\mathrm{n_{en}}} u_i N_i(\boldsymbol{x}), \tag{4}$$

where u_i are nodal values, N_i are polynomial shape functions of order p in $\boldsymbol{x}$, and $\mathrm{n_{en}}$ is the number of element nodes. Therefore, the approximation considered in NEFEM ensures reproducibility of polynomials in the physical space for any order of approximation p. See [5] for information about efficient computation of the polynomial base for any degree of interpolation and for any nodal distribution in Ω_e. The exact description of the boundary is used to perform the numerical integration on the physical subdomain Ω_e. Thus, special numerical strategies are required for every element Ω_e.

4. Numerical integration along NURBS curves

This section is devoted to the numerical integration of polynomial functions along NURBS curves. As pointed out in the introduction, the numerical integration of the constant function $f = 1$ is of particular interest in CAD. It allows to compute the

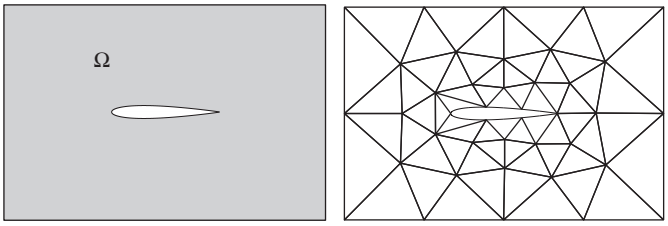

Fig. 3. Physical domain with part of the boundary defined by a NURBS curve (left) and a valid triangulation for NEFEM (right).

length of a NURBS curve and, for instance, it is useful to compute an approximate reparametrization of a NURBS curve and to subdivide the curve in equal length pieces. The numerical integration of high-order polynomials is of interest in the NEFEM context.

Given a NURBS curve parametrized by $\boldsymbol{C}$, the integral of a function f over the NURBS curve is written as

$$\int_{\boldsymbol{C}} f \, d\ell = \int_{\lambda_a}^{\lambda_b} f(\boldsymbol{C}(\lambda)) |J_{\boldsymbol{C}}(\lambda)| \, d\lambda, \tag{5}$$

where $|J_{\boldsymbol{C}}|$ denotes the norm of the differential of the NURBS parametrization. As usual, a 1D numerical quadrature is used for the numerical computation of the integral

$$\int_{\boldsymbol{C}} f \, d\ell \simeq \sum_{i=1}^{\tilde{n}} f(\boldsymbol{C}(\tilde{\lambda}_j)) |J_{\boldsymbol{C}}(\tilde{\lambda}_j)| \tilde{\omega}_j,$$

where $\tilde{\lambda}_j$ and $\tilde{\omega}_j$ are the coordinates and weights of the $\tilde{n}$ integration points in $[\lambda_a, \lambda_b]$.

Recall that the parametrization of a trimmed NURBS, $\boldsymbol{C}$, is a piecewise rational function whose definition changes at the breakpoints. Thus, an independent numerical quadrature must be considered for each one of the intervals between breakpoints in order to take into account the discontinuous nature of the parametrization. Therefore, integral of Eq. (5) is computed as

$$\int_{\boldsymbol{C}} f \, d\ell \simeq \sum_{i=q+1}^{n_k - q - 1} \left(\sum_{j=1}^{n} f(\boldsymbol{C}(\lambda_j^i)) |J_{\boldsymbol{C}}(\lambda_j^i)| \omega_j^i \right),$$

where λ_j^i and ω_j^i are the coordinates and weights of the n integration points in $[\lambda_i, \lambda_{i+1}]$.

NEFEM requires the computation of the integral of any polynomial function f over an edge of a curved element given by a trimmed NURBS, $\Gamma_e = \boldsymbol{C}([\lambda_1^e, \lambda_2^e])$. These integrals are related to the weak imposition of boundary conditions or to flux evaluation in a discontinuous Galerkin (DG) context, see for instance [5].

As discussed in the previous section, NEFEM uses polynomials directly in the physical space to approximate the solution. Therefore, the numerical integration of high-order polynomials over NURBS curves is of interest in this context. When a polynomial interpolation of degree p is considered, the integration of the weak form involves the integration of polynomials of degree less than or equal to $2p$. For instance, the computation of a mass matrix term requires the computation of integral

$$\int_{\Omega_e} N_i(\boldsymbol{x}) N_j(\boldsymbol{x}) \, d\boldsymbol{x},$$

where N_i and N_j are polynomials of degree p.

Approximations with polynomials of degree up to $p=10$ are of particular interest, see the applications presented in [5,7], requiring the numerical integration of polynomials up to degree 20.

The behavior of some well-known numerical quadratures for the computation of integrals of polynomials up to degree 20 along NURBS curves is studied through numerical examples in the next section. The selected quadratures are:

- trapezoidal and Simpson composite rules,
- Romberg's integration,
- Gauss–Legendre rules, and
- Gauss–Legendre composite rules with $n=4,8$ Gauss points in each subinterval.

Romberg's method is an improvement of the trapezoidal composite rule by using Richardson extrapolation repeatedly, see [8]. For the definition of a composite rule from a Gauss–Legendre quadrature with n Gauss points, the integration interval is decomposed in m subintervals and the simple Gauss–Legendre quadrature is used in each subinterval. The resulting quadrature has mn integration points.

4.1. Numerical tests for boundary integrals

Two different NURBS curves are considered for the numerical tests: a quarter of a circle and a portion of an airfoil, see Figs. 4 and 5. The trimmed NURBS considered for the portion of the airfoil is in fact a (polynomial) B-spline, and it corresponds to the front part of the airfoil, which is the most critical region.

Fig. 6 shows the accuracy of different numerical quadratures for computing the integral of the constant function $f=1$ and the high-order polynomial function $f=xy^{19}$ along the trimmed NURBS describing a quarter of a circle. The plots represent the relative error in the computation of the integral in Eq. (5) versus the total number of integration points using different numerical quadratures. As usual, the relative error is measured as $|I-I^\star|/|I^\star|$, where I is the approximated value of the integral using numerical quadratures and $I^\star$ is the exact value of the integral.

The use of high-order simple Gauss–Legendre quadratures is the most efficient option. Machine precision is attained with the minimum number of integration points. Composite rules from the Gauss–Legendre quadratures are also competitive. More popular composite rules, such as the trapezoidal and Simpson composite rules are not suitable for the integration along NURBS curves, due to the excessive computational cost. Even using Romber's integration more than 100 integration points are required to attain machine precision.

It is observed that the qualitative behavior of quadratures is very similar for the integration of constants and high-order polynomials. This indicates that the complexity of the integral in Eq. (5) is given by the rational definition of the NURBS and the irrational term $|J_{\boldsymbol{C}}|$, not by the degree of the polynomial to be integrated.

Fig. 7 shows the accuracy of different numerical quadratures for computing the integral of the constant function $f=1$ and the high-order polynomial function $f=x^{10}y^{10}$ along the trimmed NURBS describing the front part of the airfoil. This trimmed NURBS presents three breakpoints in its parametrization interval, and therefore, the mentioned quadratures are considered for each one of the four patches. In all plots the abscissa is the number of

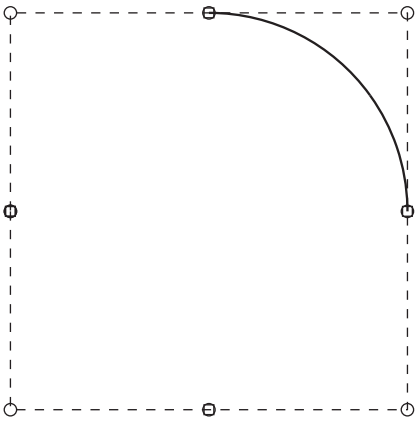

Fig. 4. Trimmed NURBS curve describing a quarter of a circle (solid line), control points (○), control polygon (dashed line) and image of the breakpoints (□).

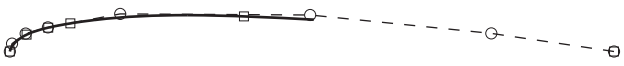

Fig. 5. Trimmed NURBS curve describing the front part of an airfoil (solid line), control points (○), control polygon (dashed line) and image of the breakpoints (□).

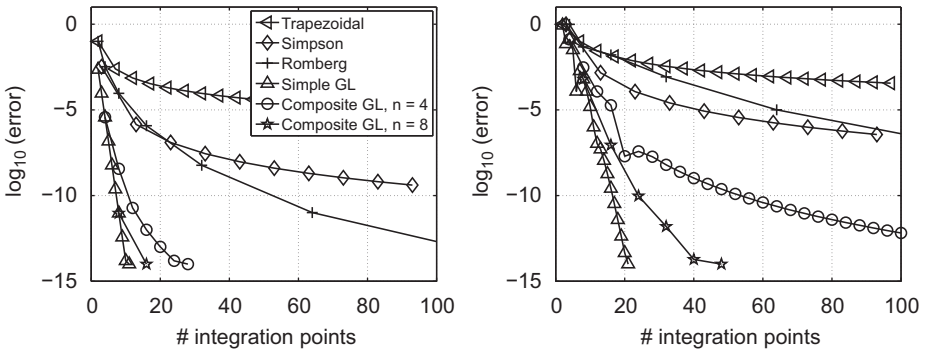

Fig. 6. Relative error for the integration of $f(x,y)=1$ (left) and $f(x,y)=xy^{19}$ (right) along the trimmed NURBS describing a quarter of a circle.

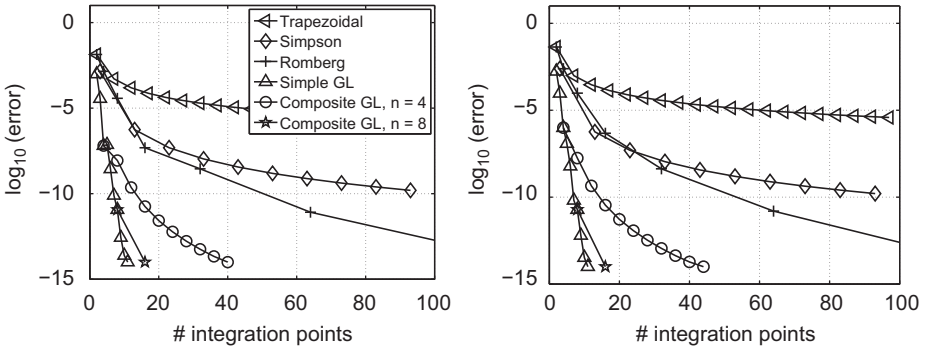

Fig. 7. Relative error for the integration of $f(x,y)=1$ (left) and $f(x,y)=x^{10}y^{10}$ (right) along the trimmed NURBS describing the front part of the airfoil.

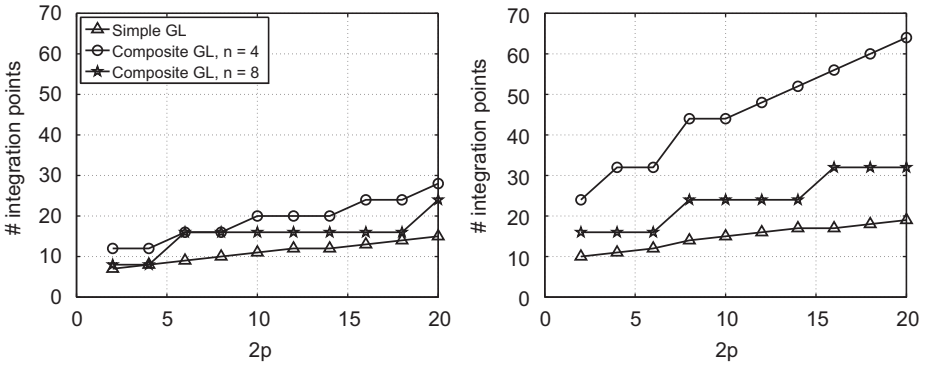

Fig. 8. Number of integration points required to integrate all the polynomials of degree of equal to $2p$ with an accuracy of 10^{-6} (left) and 10^{-10} (right) along the trimmed NURBS describing a quarter of a circle.

integration points used in each patch. Again, the most efficient option is to use simple Gauss–Legendre quadratures, and composite Gauss–Legendre quadratures with $\mathtt{n}=8$ provide very accurate results.

When a polynomial interpolation of degree p is considered in the NEFEM context, it is interesting to know the minimum number of integration points needed to integrate all the polynomials of degree less than or equal to $2p$ with a desired accuracy. Fig. 8 shows the number of integration points needed to integrate all the polynomials of a degree less than or equal to $2p$ with an accuracy of 10^{-6} and 10^{-10}, respectively, along the trimmed NURBS describing a quarter of a circle. The results using simple and composite Gauss–Legendre quadratures are displayed. For a NEFEM computation with polynomials of degree $p=5$, simple Gauss–Legendre quadratures provide an accuracy of 10^{-6} in the boundary integrals using 10 integration points. Gauss–Legendre composite quadratures with $\mathtt{n}=4$ require five subintervals, i.e. 20 integration points, to obtain the same accuracy, and Gauss–Legendre composite quadratures with $\mathtt{n}=8$ require two subintervals, i.e. 16 points. For a NEFEM computation with polynomials of degree 10, an accuracy of 10^{-6} is attained with simple Gauss–Legendre quadratures with 15 integration points, whereas composite Gauss–Legendre quadratures require 28 integration points and composite Gauss–Legendre quadratures with $\mathtt{n}=8$ require 24 integration points. If the desired accuracy is 10^{-10} the number of integration points is only slightly increased for simple quadratures, whereas composite quadratures suffer from a higher increase in computation cost. For instance, for a NEFEM computation with polynomials of degree 10, simple Gauss–Legendre quadratures require 19 points, composite quadratures with $\mathtt{n}=4$ require 16 subintervals, i.e. 64 points, and composite quadratures with $\mathtt{n}=8$ require four subintervals, i.e. 32 points. Same

conclusions are obtained for the integration along the trimmed NURBS describing the front part of the airfoil.

To conclude, numerical experiments reveal that Gauss–Legendre quadratures are very competitive for the numerical integration of polynomials along NURBS curves. Although the faster convergence is obtained for high-order simple quadratures, the use of composite rules is very attractive, allowing to control the error in a straightforward manner, see for instance [8]. In addition, it is important to note that the NURBS considered in this section do not present drastic variations of the velocity $|J_{\boldsymbol{C}}|$. Thus, a simple quadrature provides good results.

5. Numerical integration over NURBS-shaped domains

This section is devoted to the numerical integration of polynomial functions over domains with NURBS boundaries. Again, it is worth recalling that the integration of the constant function $f=1$ is of special interest in CAD because it provides the area of a NURBS-shaped domain and it is used, for instance, to compute its center of mass.

Given a domain Ω with its boundary parametrized by at least one NURBS curve, the integral of any polynomial function f over Ω can be written as

$$\int_{\Omega} f\,dx\,dy = \int_{Q} f(\Psi(\xi,\eta))|J_{\Psi}(\xi,\eta)|\,d\xi\,d\eta, \tag{6}$$

where $\Psi : Q \to \Omega$ is a parametrization of the domain and $|J_{\Psi}|$ is the determinant of its Jacobian. In general it is not trivial to define such parametrizations, specially when the boundary is given by several trimmed NURBS curves. Instead, it is easier to subdivide the domain Ω in simpler subdomains Ω_e and to perform the numerical integration over each subdomain separately. For integration purposes there are no requirements on the quality and the number of subdomains and a very coarse discretization can be used. To reduce casuistics in the implementation, this work considers triangular subdomains with no more than one edge on the boundary. Subdomains with several edges on different NURBS boundaries are split into subdomains with only one edge on a NURBS boundary. This is in fact the situation in a NEFEM integration mesh.

Two strategies to perform the numerical integration on each triangular subdomain Ω_e are proposed and compared next. The first one is based on a transformation from the straight-sided triangle I given by the vertices $\{(0,0),(1,0),(0,1)\}$, where well-known efficient triangle quadratures can be considered. The second one considers a transformation from a rectangle where quadratures are easily defined as a tensor product of 1D quadratures.

The transformation from I to Ω_e is defined by

$$\boldsymbol{\phi} : I \longrightarrow \Omega_e,$$

$$(\xi,\eta) \mapsto \boldsymbol{\phi}(\xi,\eta) := \frac{1-\xi-\eta}{1-\xi}\boldsymbol{C}(\xi) + \frac{\xi\eta}{1-\xi}\boldsymbol{x}_2 + \eta\boldsymbol{x}_3, \tag{7}$$

where $\boldsymbol{x}_1 = \boldsymbol{C}(0)$ and $\boldsymbol{x}_2 = \boldsymbol{C}(1)$ are the vertices of Ω_e on the curved boundary, and $\boldsymbol{x}_3$ is the internal vertex, see Fig. 9. Using this transformation from I to Ω_e, the contribution of Ω_e to the integral in Eq. (6) is computed as

$$\int_{\Omega_e} f\,dx\,dy = \int_{\Omega_e} f(\boldsymbol{\phi}(\xi,\eta))|J_{\phi}|\,d\xi\,d\eta \simeq \sum_{i=1}^{n_{ip}} f(\boldsymbol{\phi}(\xi_i,\eta_i))|J_{\phi}(\xi_i,\eta_i)|\tau_i, \tag{8}$$

where (ξ_i,η_i) and τ_i are, for instance, the symmetric quadrature points and weights at the triangle I, see Refs. [9,10] for further information, and $|J_{\phi}|$ is the determinant of the Jacobian of $\boldsymbol{\phi}$.

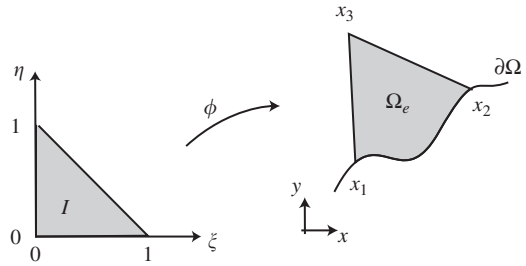

Fig. 9. Transformation $\boldsymbol{\phi}$ from the straight-sided triangle I to the subdomain Ω_e.

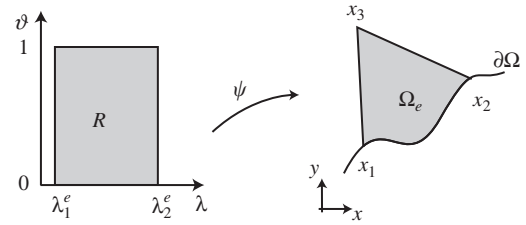

Fig. 10. Transformation $\boldsymbol{\psi}$ from the rectangle $[\lambda_1^e,\lambda_2^e]\times[0,1]$ to the curved triangle Ω_e.

The transformation from the rectangle $R = [\lambda_1^e,\lambda_2^e]\times[0,1]$ to the curved element Ω_e given by

$$\boldsymbol{\psi} : R \longrightarrow \Omega_e,$$

$$(\lambda,\vartheta) \mapsto \boldsymbol{\psi}(\lambda,\vartheta) := (1-\vartheta)\boldsymbol{C}(\lambda) + \vartheta\boldsymbol{x}_3 \tag{9}$$

is also considered, see Fig. 10. Using this transformation, the contribution of Ω_e to the integral in Eq. (6) is computed as

$$\begin{aligned}\int_{\Omega_e} f\,dx\,dy &= \int_{I} f(\boldsymbol{\psi}(\lambda,\vartheta))|J_{\psi}|\,d\lambda\,d\vartheta \\ &\simeq \sum_{i=1}^{n_\lambda}\sum_{j=1}^{n_\vartheta} f(\boldsymbol{\psi}(\lambda_i,\vartheta_j))|J_{\psi}(\lambda_i,\vartheta_j)|\omega_i\varpi_j,\end{aligned} \tag{10}$$

where (λ_i,ω_i) and $\{\vartheta_j,\varpi_j\}$ are the 1D quadrature points and weights for the intervals $[\lambda_1^e,\lambda_2^e]$ and $[0,1]$, respectively, and $|J_{\psi}|$ is the determinant of the Jacobian of the transformation $\boldsymbol{\psi}$.

Remark 1. Given a polynomial f of degree k to be integrated in Ω_e, the corresponding function to be integrated in the rectangle $[\lambda_1^e,\lambda_2^e]\times[0,1]$, after the transformation shown in Fig. 10, is $\tilde{f}(\lambda,\vartheta) = f(\boldsymbol{\psi}(\lambda,\vartheta))|J_{\psi}(\lambda,\vartheta)|$, see Eq. (10). Transformation $\boldsymbol{\psi}$ is linear in parameter ϑ and, therefore, $f(\boldsymbol{\psi}(\lambda,\vartheta))$ is a polynomial of degree k in ϑ, and $|J_{\psi}(\lambda,\vartheta)|$ is linear in ϑ. Thus, $\tilde{f}(\lambda,\vartheta)$ is a polynomial of degree $k+1$ in parameter ϑ and, in consequence, the Gauss–Legendre quadrature of order k ($k+1$ for even k) is an optimal choice for the 1D quadrature $\{\vartheta_j,\varpi_j\}$.

Note that, if $\boldsymbol{C}$ is a B-spline, i.e. a piecewise polynomial parametrization, then $\boldsymbol{\psi}$ is also a piecewise polynomial function. Therefore, the function $\tilde{f}$ is a piecewise polynomial that can be exactly integrated using Gauss–Legendre quadratures. For instance, if the boundary is described using a B-spline of degree q, interior integrals can be exactly computed with Gauss–Legendre quadratures with $p+1$ and $q(k+1)$ integration points in each direction.

Transformation in Eq. (9) allows decoupling the complexity of the NURBS direction λ and the interior direction ϑ, for which exact integration is feasible and cheap. Therefore, the evaluation of integral in Eq. (6) has similar level of difficulty as the numerical integration over a trimmed NURBS curve. This is not the case when using transformation in Eq. (7) because the two directions

are coupled and the complexity is similar to the integration over a NURBS surface.

5.1. NEFEM interior integrals

The numerical integration of the discretized weak form in the NEFEM context involves the computation of integrals of polynomial functions over an element with an edge on the NURBS boundary.

In NEFEM, the subdomains or elements Ω_e are not designed for integration purposes but for approximating the solution of the weak formulation. Therefore, some elements may have more than one edge on the NURBS boundary. In order to define the numerical quadrature these elements are split on subelements with no more than one edge defined by a NURBS boundary. Then, a different numerical quadrature is used for the computation of the integral on each subelement. It is worth remarking that subdivision is only applied to perform the numerical integration, no new degrees of freedom are introduced in such elements. In addition, some element may contain breakpoints inside its curved edges. For elements containing changes on NURBS definition on the curved edge composite quadratures to account for the discontinuity in the definition of the parametrization has to be considered. For illustration purposes the triangle with a curved edge represented in Fig. 11 is considered. The curved edge is described with a piecewise parametrization $\mathbf{C}$, whose definition changes in two points on the curved edge, marked with □.

If the transformation in Eq. (7) is used, the reference triangle should be partitioned as represented in Fig. 12, where the discontinuous lines show the changes of definition of the transformation in Eq. (7). Note that these lines originate at the breakpoints of the NURBS parametrization in the ξ axis, and are extended inside the reference element I. A composite numerical quadrature on I should be defined by using different numerical quadratures in each region. In the example represented in Fig. 12 the composite quadrature on I consists on two quadratures on the rectangular regions and one triangle quadrature.

When the transformation in Eq. (9) is considered, changes of NURBS definition are easily accommodated. The rectangle R is subdivided using the breakpoints, as represented in Fig. 13, and a numerical quadrature in R is defined only in terms of 1D quadratures. In fact, the discontinuous nature of the NURBS parametrization affects only to the λ parameter, for which a 1D composite quadrature has to be considered. The same composite quadrature used for the line integrals can used for the λ parameter. In the other parameter, ϑ, exact integration is feasible with a simple Gauss–Legendre quadrature, as discussed in Remark 1.

Remark 2. When the transformation $\boldsymbol{\psi}$ from the rectangle is considered, the integrals involved in the elemental matrices, for a NEFEM solution with interpolation of degree p, can be exactly computed for one of the parameters, ϑ, using a Gauss–Legendre quadrature with $p+1$ integration points. The numerical integration for the other direction, given by the NURBS parameter λ, presents the same difficulty as the integration over a NURBS curve, which has been commented in Section 4. Moreover, if the geometry is described with a q-th degree B-spline, the elemental matrices can be exactly computed with Gauss–Legendre quadratures with $p+1$ integration points for the ϑ parameter, and $q(p+1)$ integration points in each patch for the NURBS parameter λ.

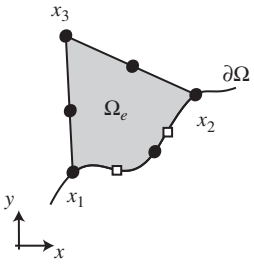

Fig. 11. Triangle with a curved edge containing changes of NURBS definition (marked with □).

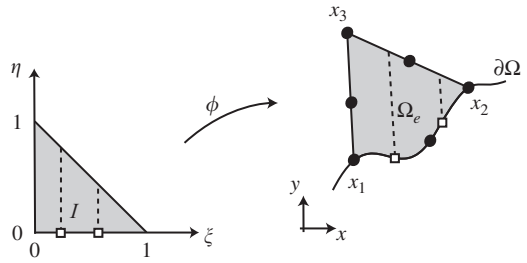

Fig. 12. Numerical integration using the transformation from I to Ω_e: subdivision of I to design a numerical quadrature taking into account changes of NURBS parametrization $\mathbf{C}(\xi)$ at points marked with □.

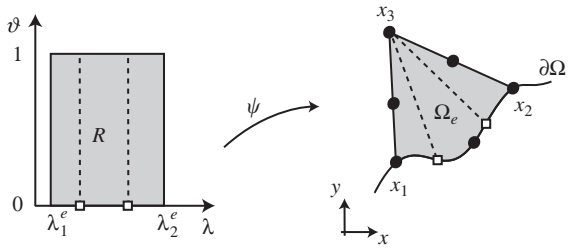

Fig. 13. Numerical integration using the transformation from R to Ω_e: subdivision of the rectangle $R=[\lambda_1^e,\lambda_2^e]\times[0,1]$ to design a numerical quadrature taking into account changes of NURBS parametrization $\mathbf{C}(\lambda)$ at points marked with □.

5.2. Numerical tests for interior integrals

The behavior of the transformations in Eqs. (7) and (9) is studied for the computation of interior integrals in a NEFEM context. When using the transformation in Eq. (7) from I to Ω_e, standard symmetric triangle quadratures are considered, see for instance [10]. When the transformation in Eq. (9) from R to Ω_e is used, simple and composite 1D Gauss–Legendre quadratures are considered in each direction, as commented in Remark 2.

Two triangular elements are considered for the numerical tests. The first one, namely Ω_1, has vertices {(1,0), (0,1), (0,0)} and one face described by the circle NURBS trimmed to the interval [0.5,0.75], see the NURBS data in Appendix A. The second triangular element, namely Ω_2 has vertices {(−0.4721,0.0277), (−0.5,0), (−0.53,0.05)} and one face described by the airfoil curve trimmed to the interval [0.9786,1.0194], see the B-spline data in Appendix A.

Fig. 14 shows the relative error for the integration of the constant function $f=1$ and the high-order polynomial $f=x^{10}y^{10}$ over the interior of the first element Ω_1. Fig. 15 shows the relative error for the integration of the low-order polynomial $f=xy$ and the high-order polynomial $f=x^{19}y$ over the interior of the second element Ω_2.

Numerical tests show the efficiency of the numerical quadratures defined using the transformation in Eq. (9) with Gauss–Legendre quadratures, especially high-order simple quadratures.

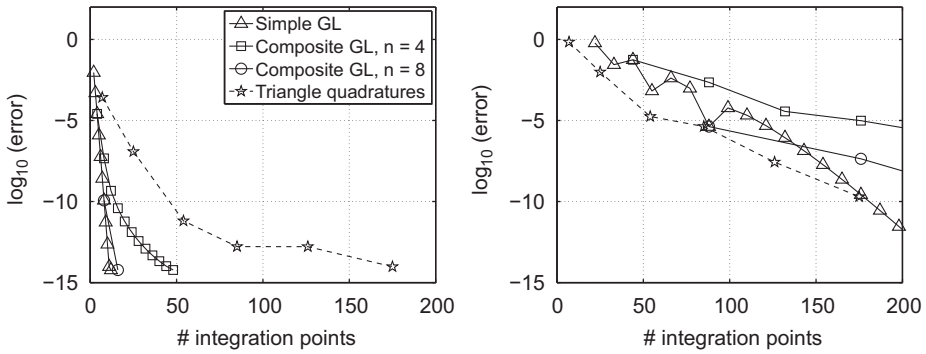

Fig. 14. Relative error for the integration of $f=1$ (left) and $f=x^{10}y^{10}$ (right) over the element Ω_1.

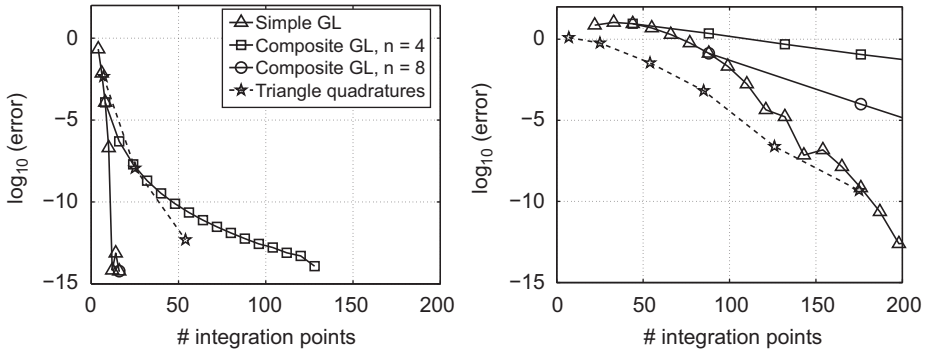

Fig. 15. Relative error for the integration of $f=xy$ (left) and $f=x^{19}y$ (right) over the element Ω_2.

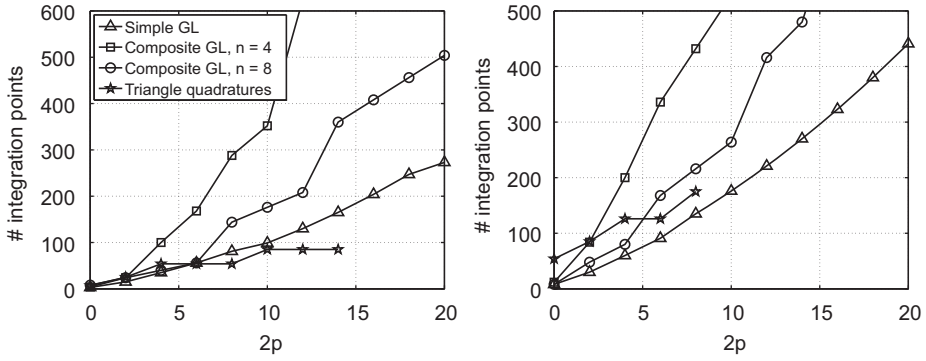

Fig. 16. Number of integration points required to integrate all the polynomials of degree of equal to $2p$ with an accuracy of 10^{-6} (left) and 10^{-10} (right) over the element Ω_1.

It is observed that triangle quadratures combined with the transformation in Eq. (7) are efficient for the integration of very high-order polynomials, see right plots in Figs. 14 and 15. Despite this advantage, the use of this strategy for the integration of low-order polynomials is far of being the most efficient option, see left plots in Figs. 14 and 15.

In a NEFEM context it is necessary to integrate all polynomial functions of degree less than or equal to $2p$ with a desired accuracy. Fig. 16 shows the total number of integration points needed to integrate with a desired accuracy all polynomials of degree less than or equal to $2p$ over the element Ω_1. A detailed view of the plot for $p \leq 4$ is depicted in Fig. 17.

It can be observed that for moderate accuracy, let say an error of 10^{-6}, simple Gauss–Legendre quadratures perform better for low-order approximations, more precisely for $p \leq 3$, whereas triangle quadratures are the most efficient option if high-order approximations are considered, see left plots in Figs. 16 and 17. It is worth remarking that for $p > 7$ the highest order symmetric quadrature rule considered here, which has order 30, is not able to provide the desired accuracy. When higher accuracy is required, let say an error of 10^{-10}, simple Gauss–Legendre quadratures are the most efficient for low- and high-order polynomials. Again, using the highest order available triangle quadrature the desired accuracy cannot be achieved for a polynomial order of approximation $p > 4$, see right plots in Figs. 16 and 17.

Fig. 18 represents the total number of integration points needed to integrate with a desired accuracy all polynomials of degree less than or equal to $2p$ over the element Ω_2. A detailed view of the plot for $p \leq 4$ is depicted in Fig. 19. Simple Gauss–Legendre quadratures perform better for low-order approximations, more precisely for $p \leq 3$, whereas triangle quadratures are the most efficient option for high-order approximations. It is worth emphasizing that the boundary of this curved elements is given by a (polynomial) B-spline.

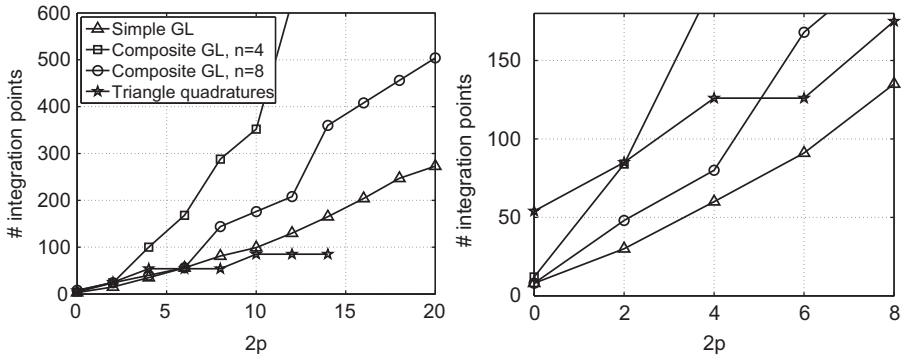

Fig. 17. Number of integration points required to integrate all the polynomials of degree of equal to $2p$ ($p \leq 4$) with an accuracy of 10^{-6} (left) and 10^{-10} (right) over the element Ω_1.

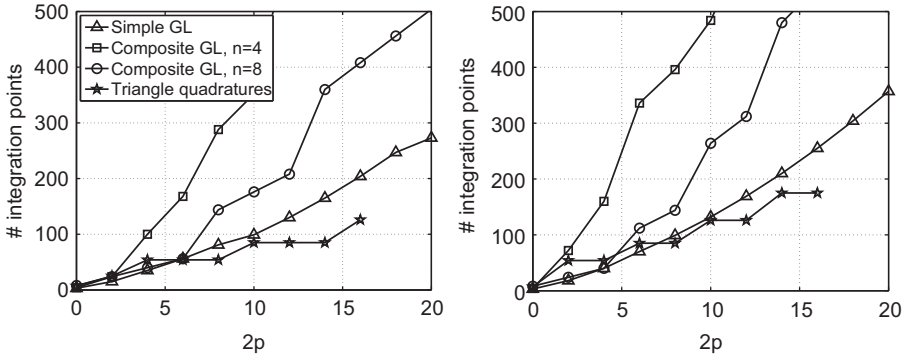

Fig. 18. Number of integration points required to integrate all the polynomials of degree of equal to $2p$ with an accuracy of 10^{-6} (left) and 10^{-10} (right) over the element Ω_2.

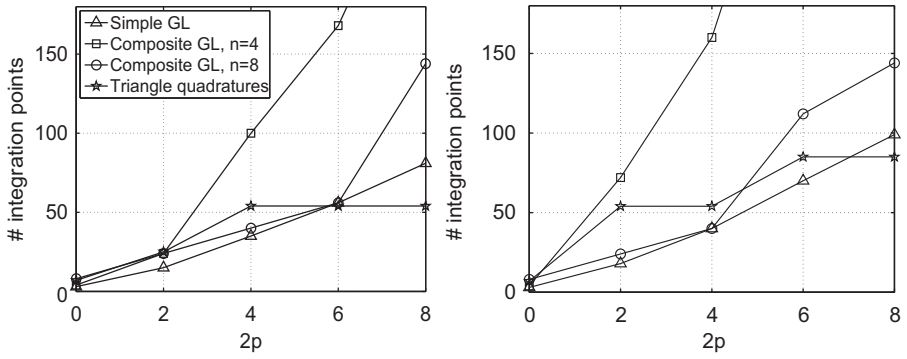

Fig. 19. Number of integration points required to integrate all the polynomials of degree of equal to $2p$ ($p \leq 4$) with an accuracy of 10^{-6} (left) and 10^{-10} (right) over the element Ω_2.

To conclude, when the boundary is given by a B-spline curve the transformation from a triangle performs better for very high-order approximations, $p \geq 4$, whereas the use of simple Gauss–Legendre quadratures performs better for low-order approximations. However, for the general case of NURBS boundaries simple Gauss–Legendre quadratures perform better for low- and high-order approximations if high fidelity is required in the numerical integration, whereas if a higher error is tolerated and high-order approximations are considered triangle quadratures are more efficient.

It is worth remarking that the mapping from a rectangle presented in this work can be easily generalized to 3D, defining a mapping from a triangular prism to a tetrahedral in the physical space. Decoupling NURBS surface directions with respect to the interior direction is feasible and efficient quadratures can be designed. The generalization of the mapping based on a triangle is obviously a mapping from a reference tetrahedral to a curved tetrahedral in the physical space, but the design of 3D quadratures when the curved face contains changes of NURBS parametrization turns out to be complex and expensive, see a detailed discussion in [11].

Finally, in order to compare the results of interior and boundary integrals, Fig. 20 presents the number of integration points in the λ direction needed to integrate with a desired accuracy all polynomials of degree less than or equal to $2p$, when using the transformation in Eq. (9). As commented earlier in this section, the complexity of the numerical integration using the transformation from a rectangle is comparable to the difficulty of the 1D numerical integration along a NURBS curve, and therefore, the convergence plots in Fig. 20 show a qualitative behavior similar to the ones presented in Section 4.

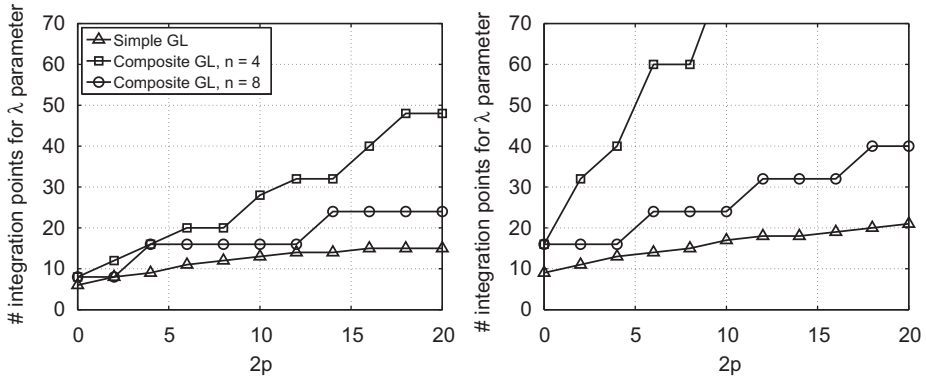

Fig. 20. Number of integration points in the λ direction required to integrate all the polynomials of degree of equal to $2p$ with an accuracy of 10^{-6} (left) and 10^{-10} (right) over the element Ω_1.

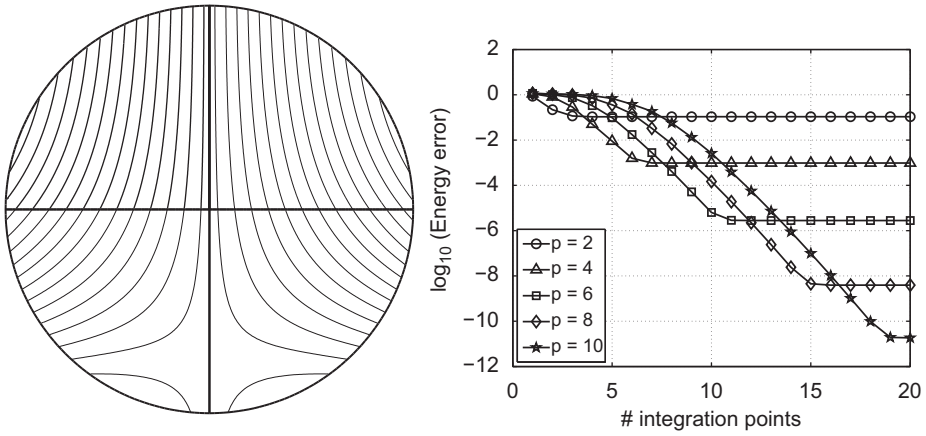

Fig. 21. Relative error for the integration over a triangular element with an edge defined by the trimmed NURBS describing the front part of the airfoil.

6. Influence of numerical integration in NEFEM computations

This section presents some numerical examples showing the influence of the number of integration points in the accuracy of NEFEM computations. A Helmholtz problem is solved using a standard continuous Galerkin formulation and more complex electromagnetic scattering applications are considered using a DG formulation.

6.1. Helmholtz problem

The following Helmholtz problem is considered:

$$\begin{cases} -\Delta u + u = s & \text{in } \Omega, \\ \nabla u \cdot \boldsymbol{n} = g_n & \text{on } \partial\Omega, \end{cases}$$

where the domain Ω is a circle of unit radius and $\boldsymbol{n}$ is the outward unit normal vector on $\partial\Omega$. The source s and g_n are such that the analytical solution of the problem is $u(x,y) = x\cos(y) + y\sin(x)$. A very coarse mesh with only four curved elements is considered, see the mesh and the isolines of the numerical solution for a NEFEM approximation with $p=10$ in the left plot of Fig. 21. Boundary integrals appearing due to the imposition of the Neumann boundary conditions are computed using simple Gauss–Legendre quadratures. The numerical integration in the element interiors is computed using the transformation in Eq. (9).

Right plot in Fig. 21 shows the solution error in energy norm as a function of the number of integration points in the NURBS parameter λ for different degrees of approximation p. Obviously, as the degree of the polynomial approximation is increased more integration points are necessary to reach maximum NEFEM accuracy in the computation. In all the experiments, the minimum number of integration points to achieve the maximum accuracy is $2p-1$. It is worth noting that the simplicity of the analytical solution and the exact boundary representation considered with NEFEM makes the numerical integration very important in this example. In fact, NEFEM needs more integration points than standard FEs to reach its maximum accuracy but as shown in the comparisons presented in [11], the results are much more accurate, two orders of magnitude more accurate than using standard isoparametric FEs.

6.2. Electromagnetic scattering

This section presents two examples involving the numerical solution of the transient Maxwell's equations applied to the computation of the scattering of a plane wave by perfect electric conductors (PECs) surrounded by free space, see [12,13].

Numerical integration is used in order to integrate the terms appearing in the weak formulation and also for the evaluation of the quantity of interest, the so-called radar cross-section (RCS), that is computed as

$$\chi(\phi) = \frac{k}{4}\left|\int_C ((n_2^C\sin\phi + n_1^C\cos\phi)E_3^f + (n_1^C H_2^f - n_2^C H_1^f))e^{jw(x\cos\phi + y\sin\phi)}\, dC\right|^2,$$

where E_l and H_l are the l-th component of the electric and magnetic fields, respectively, k is the wave number of the incident field, C is the scatterer boundary, $\boldsymbol{n}^C = (n_1^C, n_2^C)$ is the outward unit

normal to C, $j=\sqrt{-1}$, w is the angular frequency of the incident wave and the superscript f indicates the complex amplitudes of the fields evaluated in the frequency domain, see [13] for further details.

The first example considers an incident plane wave traveling in the x^+ direction and scattered by a PEC circular cylinder, which is exactly described with a quadratic NURBS curve. A coarse mesh with only four elements for the discretization of the NURBS boundary is considered, see left plot in Fig. 22. The transverse scattered field H_3 for a NEFEM solution with $p=10$ is represented in the right plot of Fig. 22, and the RCS for a NEFEM computation with $p=10$ is represented in the left plot of Fig. 23, showing an excellent agreement with the analytical solution.

The $\mathcal{L}^2([-\pi,\pi])$ norm of the relative error in the RCS is shown in the right plot of Fig. 23 for increasing number of Gauss–Legendre integration points in the λ direction. A proper integration along the NURBS boundary is crucial to obtain the NEFEM maximum accuracy. Nevertheless, it is not necessary to compute boundary integrals with machine precision: a Gauss–Legendre quadrature with $p+2$ integration points provides NEFEM maximum accuracy for this computational mesh. The transformation from a rectangle in Eq. (9) is considered for the definition of the numerical quadrature at every element with one edge given by a trimmed NURBS.

It is worth remarking that the maximum accuracy for NEFEM computation can be obtained using just $p+2$ integration points for the NURBS parameter in Eq. (10), i.e. one point more than for a standard isoparametric FE computation. As expected, if a lower number of integration points is considered, the ill-conditioning of the elemental mass matrices combined with the inaccurate evaluation of the RCS leads to clearly unacceptable results.

The last example considers an incident plane wave traveling in the y^+ direction and scattered by a PEC airfoil, which is described with a cubic B-spline curve. Again, a very coarse mesh with only four elements on the upper part of the airfoil is considered, see left plot in Fig. 24. The transverse scattered field H_3 for a NEFEM solution with $p=10$ is represented in the right plot of Fig. 24, and the RCS for a NEFEM computation with $p=10$ is represented in the left plot of Fig. 25, showing an excellent agreement with a reference solution computed in a much finer mesh.

The $\mathcal{L}^2([-\pi,\pi])$ norm of the relative error in the RCS is shown in the right plot of Fig. 25 for increasing number of Gauss–Legendre integration points in the λ direction. A proper integration along the NURBS boundary is again crucial. In this example, the maximum accuracy for NEFEM computations is obtained using just $p+1$ integration points for the NURBS parameter in (10), i.e. the same number of points than using standard isoparametric FEs.

It is worth remarking that for electromagnetic scattering applications NEFEM attains its maximum accuracy just using one integration point more than standard FEs for the circle example and exactly the same for the airfoil example. In addition, it is worth re-emphasizing that the comparison study presented in [11] shows that NEFEM provides much more accurate results, at least one order of magnitude more accurate than standard isoparametric FEs.

7. Concluding remarks

The numerical integration in domains with a NURBS boundary representation is addressed in this paper, with particular emphasis on the application to NEFEM. Numerical strategies for the integration of polynomials along trimmed NURBS curves

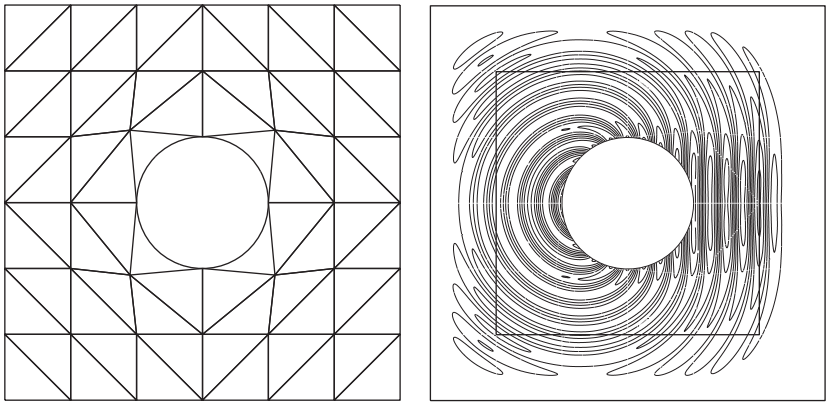

Fig. 22. Scattering by a PEC circular cylinder: computational mesh (left) and transverse scattered field for a NEFEM solution with $p=10$ (right).

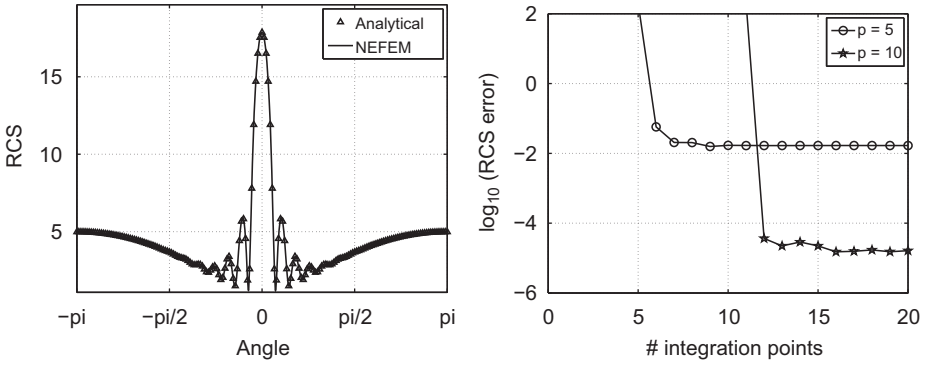

Fig. 23. Scattering by a PEC circular cylinder: RCS (left) and RCS error as a function of the number of integration points in the λ parameter (right).

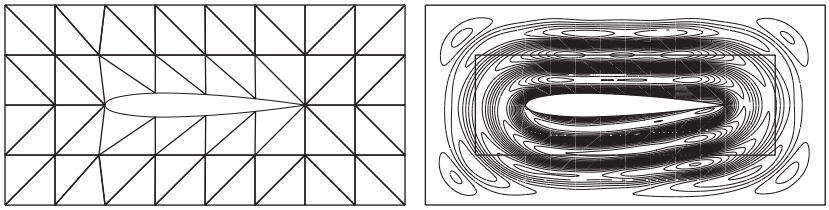

Fig. 24. Scattering by a PEC airfoil: computational mesh (left) and transverse scattered field for a NEFEM solution with $p=10$ (right).

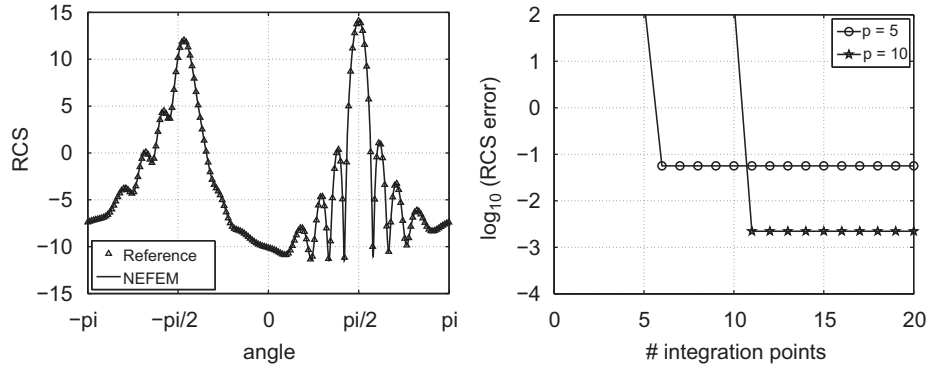

Fig. 25. Scattering by a PEC airfoil: RCS (left) and RCS error as a function of the number of integration points in the λ parameter (right).

(boundary integrals) and over elements with one edge defined by a trimmed NURBS (interior integrals) are proposed and compared through numerical examples.

Several well-known numerical quadratures are tested for the integration along trimmed NURBS curves. The numerical quadratures are considered for each one of the intervals between breakpoints, in order to take into account the discontinuous nature of the NURBS parametrization. In all numerical examples, high-order Gauss–Legendre quadratures provide the required precision with the lowest computational cost. Nevertheless, composite rules from Gauss–Legendre quadratures are also very competitive, and allow the definition of adaptive strategies to ensure the required precision for any trimmed NURBS and any polynomial.

Two transformations are compared for the definition of numerical quadratures in an element with one edge defined by a trimmed NURBS. The first transformation defined from a rectangle allows to decouple the complexity of the NURBS parameter with respect to the interior direction The numerical integration of NEFEM elemental matrices for a 2D curved element can be exactly computed for one of the parameters with $p+1$ integration points for a p-th degree polynomial interpolation. The complexity of the numerical integration for the other direction, given by the NURBS parameter, is comparable to the difficulty of the 1D integration of a polynomial over a NURBS curve, and therefore, the previously tested 1D quadratures can be used. The second transformation, defined from a triangle where efficient quadratures are considered, performs better in the particular case of a (polynomial) B-spline boundary or when high-order approximations are considered but no high accuracy in the numerical integration is mandatory. When high fidelity in the numerical integration is required, the transformation from the rectangle turns out to be very efficient.

The numerical solution of the Helmholtz equation and the transient Maxwell's equations, for the computation of the RCS of a wave scattered by a PEC object, reveals that a proper integration along the NURBS boundary is crucial to obtain the NEFEM maximum accuracy. However, it is worth noting that the maximum NEFEM accuracy can be reached with a reasonable amount of integration points, in some applications just the same number of integrations points used by standard isoparametric FEs.

Table A1
Control points and weights for the unit circle.

i	$\boldsymbol{B}_i$	v_i
1	$(-1, 0)$	1
2	$(-1, -1)$	$\frac{\sqrt{2}}{2}$
3	$(0, -1)$	1
4	$(1, -1)$	$\frac{\sqrt{2}}{2}$
5	$(1, 0)$	1
6	$(1, 1)$	$\frac{\sqrt{2}}{2}$
7	$(0, 1)$	1
8	$(-1, 1)$	$\frac{\sqrt{2}}{2}$
9	$(-1, 0)$	1

Table A2
Control points for upper part of the airfoil.

i	$\boldsymbol{B}_i$
1	(0.5000, 0.0000)
2	(0.2971, 0.0299)
3	(−0.0029, 0.0605)
4	(−0.3172, 0.0622)
5	(−0.4368, 0.0399)
6	(−0.4737, 0.0298)
7	(−0.4962, 0.0131)
8	(−0.5000, 0.0000)

Acknowledgments

This work was partially supported by the *Ministerio de Ciencia e Innovación* under grant CGL2008-06003-C03-02 and *AGAUR, Generalitat de Catalunya* under grant 2009 SGR 875.

Appendix A. Control data for NURBS objects

A.1. Circle NURBS

There are many options to define a NURBS describing a circle. A commonly used options is to define a quadratic NURBS with four rational segments. The knot vector is

$\Lambda = \{0,0,0,0.25,0.5,0.75,1,1,1\}$,

and the control points and weights are detailed in Table A1.

A.2. Airfoil B-spline

Some airfoils have analytical expressions but in the context of airfoil shape optimization, it is usual to work with an approximation using B-splines. The airfoil considered in this work is an approximation of the NACA0012 airfoil, see [14], defined using the knot vector

$\Lambda = \{0,0,0,0,0.6153,0.9037,0.9409,0.9786,1.0194,1.0194,1.0194,1.0194\}$,

and eight control points, see Table A2. Recall that B-splines are a particular case of NURBS, where all control weights are equal, and therefore only the control points have to be specified.

References

[1] L. Piegl, W. Tiller, The NURBS Book, Springer-Verlag, London, 1995.
[2] F. Bassi, S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, J. Comput. Phys. 138 (2) (1997) 251–285.
[3] D. Xue, L. Demkowicz, Control of geometry induced error in hp finite element simulations. I. Evaluation of FE error for curvilinear geometries, Int. J. Numer. Anal. Model. 2 (3) (2005) 283–300.
[4] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Comput. Methods Appl. Mech. Eng. 194 (39–41) (2005) 4135–4195.
[5] R. Sevilla, S. Fernández-Méndez, A. Huerta, NURBS-enhanced finite element method (NEFEM), Int. J. Numer. Methods Eng. 76 (1) (2008) 56–83.
[6] R. Sevilla, S. Fernández-Méndez, A. Huerta, 3D NURBS-enhanced finite element method (NEFEM), Int. J. Numer. Methods Eng., in press, doi:10.1002/nme.3164.
[7] R. Sevilla, S. Fernández-Méndez, A. Huerta, NURBS-enhanced finite element method (NEFEM) for Euler equations, Int. J. Numer. Methods Fluids 57 (9) (2008) 1051–1069.
[8] P. Davis, P. Rabinowitz, Methods of Numerical Integration, second ed., Academic Press, 1984.
[9] J.N. Lyness, D. Jespersen, Moderate degree symmetric quadrature rules for the triangle, IMA J. Appl. Math. 15 (1975) 19–32.
[10] S. Wandzura, H. Xiao, Symmetric quadrature rules on a triangle, Comput. Math. Appl. 45 (12) (2003) 1829–1840.
[11] R. Sevilla, S. Fernández-Méndez, A. Huerta, Comparison of high-order curved finite elements, Int. J. Numer. Methods Eng., in press, doi:10.1002/nme.3129.
[12] C.A. Balanis, Advanced Engineering Electromagnetics, John Wiley and Sons, New York, 1989.
[13] A. Taflove, Computational Electrodynamics: The Finite-Difference Time-Domain Method, Artech House, Inc, 1995.
[14] C.L. Ladson, C.W. Brooks, A.S. Hill, D.W. Sproles, Computer program to obtain ordinates for naca airfoils, Technical Report, NASA TM-4741, NASA Langley Research Center, 1996. 〈http://techreports.larc.nasa.gov/ltrs/PDF/NASA-96-tm4741.pdf〉.